A Homomorphic Encryption Scheme Over Integers Based on Carmichael's Theorem

Pramod Kumar Siddharth
JamiaMilliaIslamia University,New Delhi
siddharthpramodkr@gmail.com

Om Pal
Ministry of Electronics and IT, Govt. of India
ompal.cdac@gmail.com

Bashir Alam
Department of Computer Engineering,
JamiaMilliaIslamia New Delhi
babashiralam@gmail.com

Abstract— **The homomorphic properties of the cryptographic techniques take the attention of the scholars and makes it open research problem. The traditional encryption schemes don't support the operations to be computed on the encrypted data, which may compromise the privacy of the sensitive data. The homomorphic encryption supports the algebraic operations to be computed on the encrypted data. This property of the homomorphic encryption scheme have a wide application areas such as secure electronic voting, multiparty computation, private searching, delegation of computation and many more. In this paper we proposed a homomorphic encryption scheme based on the Carmichael's theorem over integers. The operations in the scheme are modular arithmetic. The paper also discuss the security scheme and further optimization are pointed out.**

Keywords: probabilistic, homomorphism, homomorphic encryption, Charmichael.

I. INTRODUCTION

When the data is outsourced to the cloud computing or distributed computing the privacy of the sensitive data can be compromised and the possible issues and attacks which can compromise the privacy are: untrusted third party computation, side channel attacks and the implementation bugs in the system. Due to these issues an encryption scheme is required to allow the algebraic operations to be computed on the encrypted data without prior knowledge of the original message and disables the decryption at the untrusted third party site. The traditional encryption systems do not allow such computation without decryption of the data and can't be no longer used in the cloud computing or distributed computing.

Then the concept of the privacy homomorphism[1] (homomorphic encryption) is introduced by the Rivest, Adleman and Dertouzos, which supports the computations on the encrypted data.

To understand homomorphic encryption, it is essential to understand the algebraic meaning of the phrase. We get the term homomorphic from the algebraic term homomorphism, which refers to a mapping between two groups $(\mathbb{G}, \otimes)$ and $(\mathbb{H}, \oslash)$.

Let $(\mathbb{G}, \otimes)$ and $(\mathbb{H}, \oslash)$ are two groups over some algebraic operation. A function $f: \mathbb{G} \Rightarrow \mathbb{H}$ is called homomorphism. If $f(x \otimes y) = f(x) \oslash f(y)$ for all $x, y \in \mathbb{G}$, that is, if f commutes with the group operations of $\mathbb{G}$ and $\mathbb{H}$. A group homomorphism $f: \mathbb{G} \Rightarrow \mathbb{H}$ is called isomorphism, if there exists a group homomorphism $f: \mathbb{H} \Rightarrow \mathbb{G}$, such that

a. $g \otimes h = (id_{\mathbb{G}})$ and
b. $h \otimes g = (id_{\mathbb{G}})$

$f(x \otimes y) = f(x) \otimes f(y)$

for all $x, y \in \mathbb{G}$ and $f(x), f(y) \in \mathbb{H}$

This notion can then be extended to rings or similar algebraic objects in the same category with multiple operations. In[1], Rivest presented the privacy homomorphism system in the form of algebraic system, which can be explained as: An algebraic system consists of a set S,

a. some operations $f_1, f_2, f_3, \ldots$
b. some predicates $p_1, p_2, p_3, \ldots$ and
c. some distinguished constants $s_1, s_2, s_3, \ldots$

The system is denoted as $\{S; f_1, f_2, f_3, \ldots; p_1, p_2, p_3, \ldots; s_1, s_2, s_3, \ldots\}$. For example, the system consisting of the integers under the usual set of operations, denoted $\{\mathbb{Z}; +, -, \times, \div; \leq; 0, 1\}$, where $\mathbb{Z}$ is the set of integers. Let's have two algebraic systems $\mathbb{C}$ and $\mathbb{U}$ given as:

$\mathbb{U} = \{S; f_1, f_2, \ldots, f_r; p_1, p_2, \ldots, p_h; s_1, s_2, \ldots, s_v\}$ and

$$\mathbb{C} = \left\{ \begin{matrix} S'; f'_1, f'_2, \ldots, f'_r; p'_1, p'_2, \ldots, p'_h; \\ s'_1, s'_2, \ldots, s'_v \end{matrix} \right\}$$

The encoding and decoding map elements from $\mathbb{U}$ to $\mathbb{C}$ and vice versa. The decoding function is $\phi: S' \Rightarrow S$ and

encoding function is $\phi^{-1}: S \rightarrow S'$. The user's database is denoted as the sequence $d_1, d_2, \ldots$ each d_i is element of S. The user encodes each datum before giving to the system: the encoded database is $\phi^{-1}(d_1), \phi^{-1}(d_2), \ldots\ldots$

To operate on the encrypted data, the decoding function ϕ must be homomorphism from $\mathbb{C}$ to $\mathbb{U}$, such that

$$(\forall i)(a, b, \ldots)\left[f'_i(a, b, \ldots) = c \rightarrow f^r(\phi(a), \phi(b), \ldots) = \phi(c)\right]$$

and

$$(\forall i)(a, b, \ldots)p^r(a, b, \ldots) = p(\phi(a), \phi(b) \ldots)$$

And $(\forall i)\phi(S'_i) = S_i$. The decoding function φ caries each operation in $\mathbb{C}$ into the corresponding operation in $\mathbb{U}$. If user wants to know the value of $f_1(d_1, d_2)$, then the system computes $f^r{}_1(\phi^{-1}(d_1), \phi^{-1}(d_2))$. Since this is homomorphism

$$\phi\left(f^r{}_1(\phi^{-1}(d_1), \phi^{-1}(d_2))\right) = f_1(d_1, d_2)$$

The requirements on the choice of the $\mathbb{C}$ and functions ϕ and $\phi^{-1}$are:

a. The ϕ and ϕ^{-1} should be easy to compute.
b. The f_i and p_i in $\mathbb{C}$ should be efficiently computable.
c. The expansion should be minimum.
d. Knowledge of $\phi^{-1}(d_i)$for many data $d_i$should not be sufficient to reveal ϕ.
e. Knowledge of d_i and $\phi^{-1}i,d_i)$for several values of d_i should not be sufficient to reveal ϕ.
f. The operations and predicates in $\mathbb{C}$should not be sufficient to yield an efficient computation of ϕ.

In this paper we represent the algebraic homomorphic encryption system over the integers, which is based on the Carmichael's[5] theorem. The scheme is probabilistic scheme in which the encryption of the same message depends on some randomized integer and produces the different ciphertexts each time encrypted.

II. HOMOMORPHISM

The idea of homomorphism was first proposed in 1978 by Rivest, Adleman and Dertouzos in their paper —On Data Banks and Privacy Homomorphisms[1]. Homomorphic encryption is the encryption function which allows the encrypted data to be operated without knowledge of the decryption function (original data). For plaintexts P_1 and P_2 and corresponding ciphertextC_1 and C_2, a homomorphic encryption scheme permits meaningful computation of $P_1 \circledcirc P_2$ from C_1 and C_2 without revealing P_1or P_2.The cryptosystem may be additive or multiplicative homomorphism depending upon the algebraic operation $\circledcirc$ which can be addition or multiplication**.**

Definition: An encryption scheme is said to homomorphic with respect to some $\circledcirc$ operation on P if we have:

$$Decrypt\left(Encrypt(P_1) \circledcirc Encrypt(P_2)\right) = Decrypt(Encrypt(P_1 \circledcirc P_2)) = P_1 \circledcirc P_2$$

An additive homomorphic encryption is the encryption function in which the decryption of a sum of ciphertexts is the sum of the corresponding messages. That is

$$Decrypt\left(Encrypt(P_1) \oplus Encrypt(P_2)\right) = Decrypt(Encrypt(P_1 + P_2)) = P_1 + P_2$$

A multiplicative homomorphic encryption is the encryption function in which the decryption of a product of ciphertexts is the product of the corresponding messages. That is

$$Decrypt\left(Encrypt(P_1) \otimes Encrypt(P_2)\right) = Decrypt(Encrypt(P_1 \times P_2)) = P_1 \times P_2$$

III. PROPOSED SCHEME

We proposed a Homomorphic Encryption scheme, which supports addition and multiplication on the encrypted data. The proposed algorithm is probabilistic encryption scheme in which a single message is encrypted into different ciphertext each time the message is encrypted with the same encryption key. A random integer r is used to randomize the encryption function. Our proposed algorithm is developed for positive integers. Our algorithm is based on the large prime numbers and the Carmichael's[5] theorem. The Carmichael's theorem.

A. OVERVIEW OF THE SCHEME

Let we have two large prime numbers p and q, and product of these numbers is calculated as $n = p \times q$ and the Carmichael's function $\lambda(n) = lcm\left((p-1) \times (q-1)\right)$. Now for any $w \in \mathbb{Z}_n^*$ the following equations holds:

$$1 \equiv w^{n\lambda} \bmod n^2 \text{ and}$$

$$1 \equiv w^{\lambda} \bmod n$$

We use both of this theorems for enciphering and deciphering and the cyclic property of Carmichael's function used, which is stated as:

$$w^{\lambda+1} = w \mod n$$

The proposed HE algorithm is given as:

1. Select two large prime integers p and q and compute $n = p \times q$, and Carmichael's function λ is evaluated as: $\lambda(n) = (p-1, q-1)$.
2. $\lambda(n)$ and n is used for encryption and n is used for decryption.

3. For the encryption of message $0 \ll m \ll n$ and a random integer r .The message is encrypted as:

$$c = m^{r^{\lambda}+1} \bmod n^2$$

4. The decryption function of the algorithm is given as:

$$m = c \bmod n$$

B. CORRECTNESS

The correctness of the scheme is given as:

$c = m^{r^{\lambda}+1} \bmod n^2$

$D(c) = c \bmod n$

$$D(c) = (m^{r^{\lambda}+1} \bmod n^2) \bmod n$$

Here in the above equation we can change the order of the modulus because n is raised to n^2 and this will make no effect on the remainder and we can write it as below and manipulate further.

$$D(c) = (m^{r^{\lambda}+1} \bmod n \ ) \bmod n^2$$

from the theorem $m^{\lambda} \bmod n = 1$any power raised to m^{λ} will produce the same remainder produced by m^{λ}, then for $m \ll n$

$D(c) = ((1)(m \mod n) \bmod n^2)$

$D(c) = (m)$

Decryption of ciphertext$c$ produces the plaintext message m.

C. HOMOMORPHISM

Our proposed scheme is homomorphic scheme, which supports both the additive and multiplicative homomorphism property. For messages $m_1$and m_2 ,we have the corresponding ciphertexts as c_1 and c_1, and random integers r_1 and r_1 used for deciphering respectively. The multiplicative and additive homomorphicproperty, and their proofs are presented below.

D. MULTIPLICATIVE HOMOMORPHISM

E.

Multiplicative homomorphism is stated as:

$m_1, m_2 = Dec[Enc(m_1), Enc(m_2)] = Dec[c_1, c_2]$

Dec is decryption and Enc is encryption function.

Proof: The property is given as:

$m_1, m_2 = Dec[Enc(m_1), Enc(m_2)] = Dec[c_1, c_2]$

$Dec[c_1, c_2]$

$= ((m_1{r_1}^{\lambda+1} \bmod n^2), (m_2{r_2}^{\lambda+1} \bmod n^2)) \bmod n$

We can rewrite as:

$m_1, m_2 = ((m_1{r_1}^{\lambda+1}, m_2{r_2}^{\lambda+1}) \bmod n \ ) \bmod n^2$

The same steps of decryption function can be taken to prove the correctness of multiplicative property of the scheme and we have:

$= ((m_1 \ . m_2 \ ) \bmod n \ ) \bmod n^2$

For m_1 and$m_2 \ll n$

$= (m_1 \ . m_2 \ ) \bmod n^2$, and

$= m_1 \ . m_2$

F. ADDITIVE HOMOMORPHISM

Additive homomorphism is stated as:

$m_1 + m_2 = Dec[Enc(m_1) + Enc(m_2)] = Dec[c_1 + c_2]$

Proof:

$m_1 + m_2 = Dec[Enc(m_1) + Enc(m_2)] = Dec[c_1 + c_2]$

$Dec[c_1 + c_2] = ((m_1{r_1}^{\lambda+1} \bmod n^2) + (m_2{r_2}^{\lambda+1} \bmod n^2)) \bmod n$

We can also prove the correctness of the additive property of the scheme.

$= (m_1 \ ) \bmod n^2 + (m_2 \ ) \bmod n^2$

$- m_1 \ + m_2$

The scheme is best suitable for integers.

IV. DELEGATION COMPUTATION

Lets take an example of delegation of computation where the user doesn't have the required resources to perform the computation on the data. In this case the user outsource his/her data to some service provider over the network or to the cloud. Now the user follow the following operation to perform the computation on his/her data.

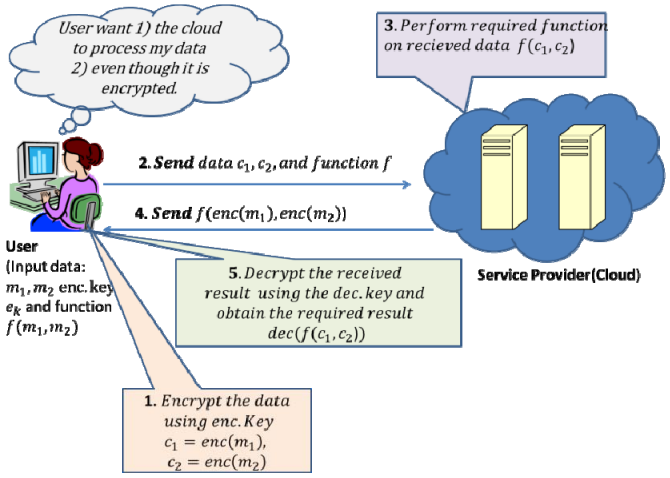

Figure 1. Delegation of Computation

User have two integer values $m_1 = 8$ and $m_2 = 7$. He/she wants to perform some operation on the available data(addition/multiplication). For this reason user encrypt his/her data and send it to service provider along with the function to be performed. The figure 1 given above depicts the all steps involved in the delegation of computation.

User selects two prime integers $p = 11$ and $q = 13$ to obtain the encryption and decryption key then hecalculate $n = p \times q = 11 \times 13 = 143$, and $\lambda(n) = (p - 1, q - 1) = (10, 12) = 60$.

After the selection of the prime numbers the user selects two random integers to randomize the ciphertext.

1. $m_1 = 8$ and $r_1 = 9$,and
2. $m_2 = 7$ and $r_2 = 5$

The encryption is performed as:

$c_1 = m_1^{r_1 d+1} mod\ n^2$

$= 8^{9*69+1} \bmod 20449$

$= 15733$

$c_2 = m_2^{r_2 d+1} mod\ n^2$

$= 7^{5*69+1} \bmod 20449$

$= 1723$

The user sends the these two encrypted numbers along with the function to the service provider. The service provider receives the number in encrypted format and is not authorized to decrypt the numbers. The service provider perform the required operation on the data as follows as:

Multiplication is computed as:

$MUL[c_1, c_2] = 1723 * 15733 = 26950629$

Addition is computed as:

$ADD[c_1 + c_2] = 1723 + 15733 = 17456$

After performing the required operations on the data, the service provider send back the result to the user and then on receiving the result from the service provider the user decrypt the data to obtain the actual result as follows:

Multiplication is obtained as:

$m_1 . m_2 = 26950629\ mod\ 148$

$= 21$

Addition is computed as:

$m_1 + m_2 = 17456\ mod\ 148$

$= 10$

After obtaining the actual result by decryption the user can use the numbers.

V. SECURITY OF THE SCHEME

The security [9] of this scheme is based on large prime number. This scheme has the encryption function implements the trapdoor function: for a given message m we can compute a function f such that $x = f(m)$, no function exists by which we can evaluate m form x such that $m = f^{-1}(x)$. The scheme is probabilistic scheme: for a given message m, we have a random integer, which randomize the ciphertext each time message is encrypted. The scheme is secure against the chosen ciphertext attack. The major drawback is the decryption function, if an adversary get the any information about n then the message can be recovered easily. The security of the decryption function must have to enhance and development of a complex decryption function is to be designed.

VI. APPLICATION AREAS

The proposed scheme is both additive as well as multiplicative homomorphic. The scheme can be used in secure electronic voting system[3] due to its additive homomorphism. This scheme can also be used in multiparty computation[3], delegation of computation and searching over encrypted data.

VII. CONCLUSION

We have proposed an HE scheme based on the Carmichael's theorem and the cyclic property of the Carmichael's function. The scheme is probabilistic in nature and uses a randomized integer for the encryption function. The scheme is best suitable for the integers.

References

[1] R. Rivest, L. Adleman, and M Dertouzos, "On data banks and privacy homomorphisms", in Foundations of Secure Computation, pp. 169–177, Academic Press, 1978.

[2] Brickell and Y. Yacobi, "On privacy homomorphisms", in Advances in Cryptology (EUROCRYPT '87), vol. 304 of Lecture Notes in Computer Science, pp. 117–126, Springer, New York, NY, USA, 1987.

[3] Gerhard Potzelsberger, "KV Web Security: Applications of Homomorphic Encryption", May 23, 2013

[4] Caroline Fontaine and Fabien Galand, Review Article "A Survey of Homomorphic Encryption for Nonspecialists" CNRS/IRISA-TEMICS, Campus de Beaulieu, 35042 Rennes Cedex, France, 24 October 2007

[5] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in Advances in Cryptology (EUROCRYPT '99), vol. 1592 of Lecture Notes in Computer Science, pp. 223–238, Springer, New York, NY, USA, 1999."

[6] C. Gentry, "Fully homomorphic encryption using ideal lattices", Symposium on the Theory of Computing (STOC), 2009, pp. 169-178

[7] Craig Gentry and ShaiHalevi, "Implementing Gentry's fully-homomorphic encryption scheme," Advances in Cryptology–EUROCRYPT 2011, pp. 129– 148, 2011.

[8] Van Dijk, Marten, Craig Gentry, ShaiHalevi, and VinodVaikuntanathan. "Fully homomorphic encryption over the Integers". Advances in Cryptology EUROCRYPT 2010 (2010): 24-4

[9] Yu Yu, JussipekkaLeiwo, Benjamin Premkumar, "A Study on the Security of Privacy Homomorphism", Nanyangchnological University, School of Computer Engineering, Proceedings of the Third International Conference on Information Technology: New Generations (ITNG'06), IEEE 200